

# Analog Control Types

*Channel device control*

Sep 22, 1989

The Local Station software supports a variety of hardware interfaces. Part of the support is that given to analog channel settings. The information in the analog control field of the analog descriptor for a given channel device describes the specific form of control which is to be used for that channel. This note details the various forms of the analog control field for each type. If the setting action appears successful (no bus error or other errors), the setting word is updated with the data value for the given channel.

The analog control field consists of 4 bytes. The first byte is the analog control type, a small index value. The meaning of the other 3 bytes depends upon the type byte. The types currently supported are:

- 00 No analog control (parameter page will not mark it with a “-”)
- 01 Datel Multibus D/A (used in Linac)
- 02 Motor (setting value is desired reading, relative setting is #steps)
- 03 Bipolar multiplex D/A (used in Linac)
- 04 Unipolar multiplex D/A (used in Linac)
- 05 Memory word (accessed as two bytes)
- 06 i8253 timer
- 07 M6840 timer
- 08 1553 D/A (12-bit)—used in rack monitor
- 09 Analog Devices RTI-602 D/A board
- 0A Memory word (accessed as one word)
- 0B Message queue setting to another cpu (co-processor)
- 0C Unsigned 12-bit D/A (in short I/O space)
- 0D Burr-Brown MPV904 12-bit D/A board
- 0E 1553 D/A (16-bit)
- 0F AMD9513 timer (32-bits from pair of channels)
- 10 Memory byte (single byte no shift)
- 11 Memory byte (single byte w/ shift in short I/O space)

## Analog control field formats



No analog control (parameter page will not mark it with a “-”)



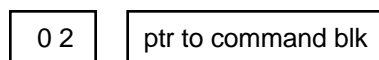
Datel Multibus D/A (used in Linac)

The ch# byte is the board's channel#. The addr word is the board's address (sign-extended). No such hardware in VME systems.



Memory-mapped motor (used in Linac)

The bit# in the range 0–7 is the bit in the byte addressed by the 16-bit address (sign-extended). Motor pulses (~20  $\mu$  sec hi-active pulses) are formed at a 150 Hz rate driven by an interrupt from a 68901 timer on the crate utility board.



1553-based motor

The ptr must be above \$100000 to distinguish this case from the above memory-mapped case. (This is no problem with 1-Mbyte of memory on the cpu board based at \$000000.) The command block houses the 1553 command for a single word write of the two's-complement #steps to be issued to the motor. The external hardware is expected to deliver the pulses. The 150 Hz interrupt that is used for the memory-mapped case decrements a counter in order to provide a shadow of the countdown register. This can be used to determine whether the motor is still running, but it assumes that the external hardware's motor pulse rate is also 150 Hz, which may not be the case.



Bipolar multiplex D/A (used in Linac)

The ch# byte includes the chassis# 0–7 and the channel# 0–15 concatenated to form a 7-bit value. The addr is sign-extended to form the address of the D/A.



Unipolar multiplex D/A (used in Linac)

This is the same as type#3, but the setting value is clamped to zero if negative.

0 5	ptr to memory
-----	---------------

Memory word (accessed as two bytes)

The 24-bit memory address is mapped into a 32-bit address as follows:

If the address is < \$F00000, the hi byte of the 32-bit address is \$00.

If the address is \$F00000, the hi byte of the 32-bit address is \$FF.

This mapping scheme allows entry of short I/O addresses plus all addresses below 15 Mbytes. The data word is written as two consecutive bytes.

0 6	ch#	addr
-----	-----	------

Intel 8253 timer

The ch# is in the range 0–2. The addr is sign-extended to form the address of the 8253 chip. (Not used in VME system)

0 7	—	addr
-----	---	------

Motorola 6840 timer

The addr ends in 2, 4 or 6 to indicate which of three 16-bit timer channels is to be set. Setting values 0 are clamped to \$0001. (Not used in VME system)

0 8	ptr to command blk
-----	--------------------

1553 D/A (12-bit)—used in rack monitor

The command block houses the 1553 command for a single word write to the D/A.

0 9	address of board
-----	------------------

Analog Devices RTI-602 D/A board

The address is mapped into 32-bits via the type#5 scheme. The data value is converted to offset binary and written to the resultant address.

0 A	ptr to memory
-----	---------------

Memory word (accessed as one word)

The address is mapped into 32-bits via the type#5 scheme. The data word is written as one 16-bit word. (Some hardware boards require this.)

0 B	type	index
-----	------	-------

Message queue setting to another cpu (co-processor)

An analog control message is placed into a co-processor message queue. The cpu# is included in bits 6-4 (mask=\$70) of the type byte. The message placed into the queue is formatted in this way:

size=8
type & \$8F
index
data

The first word is the size of the message, which in this case is always 8 bytes. The second word is the type byte anded with \$8F to remove the cpu#. The third word is an index which may have any value and serves, in conjunction with the type value, to identify what is controlled to the co-processor cpu. Now the message queue to a co-processor is more general than this use for analog control. Other message types may be passed to the cpu by the use of setting commands that use listype #45. The type word used in those messages should not conflict with those used here. (One should not use listype #45 to send the same message as is used for analog control, because the setting word associated with the analog channel will not get updated.) An easy way to insure there is no conflict is to use type word values \$100, since these analog control messages use index words \$8F.

0 C	—	addr
-----	---	------

Unsigned 12-bit D/A (in short I/O space)

The addr is assumed to be in short I/O space. The data word is clamped to zero if negative.

0 D	address of board
-----	------------------

Burr-Brown MPV904 12-bit D/A board

The address is mapped into 32-bits by the same scheme used in type #5. The data value is converted to complement offset binary and right-justified to match what the board expects.

0 E	ptr to command blk
-----	--------------------

1553 D/A (16-bit)

This is the same as type #8, but knob control sensitivity (as used with listype#7) is based upon 16 bits rather than 12.

0 F	code	addr
-----	------	------

AMD9513 timer (32-bits from pair of channels)

The addr is taken to be in short I/O space. The code values are of 4 types:

- 0x Coarse channel
- 1x Fine channel
- 2x Clock event selection
- 3x Clear all events

The x nibble is used to identify the timer channel (0–7 range) that is to be controlled on the clock timer board. More details on this can be found in the document entitled “VME Clock Timer Board.”

\$1 0	ptr to memory
-------	---------------

Memory byte (single byte no shift)

The lo byte of the data word is written to the given address (mapped to 32 bits using the type #5 scheme).

\$1 1	shift	addr
-------	-------	------

Memory byte (single byte w/ shift in short I/O space)

The shift count is used to right shift the data word before writing the lo byte to the given address which is assumed to be in short I/O space.